

## **Burst Balloons** [\(View\)](#)

You are given  $n$  balloons, indexed from  $0$  to  $n - 1$ . Each balloon is painted with a number on it represented by an array `nums`. You are asked to burst all the balloons.

If you burst the  $i^{\text{th}}$  balloon, you will get `nums[i - 1] * nums[i] * nums[i + 1]` coins. If  $i - 1$  or  $i + 1$  goes out of bounds of the array, then treat it as if there is a balloon with a `1` painted on it.

Return *the maximum coins you can collect by bursting the balloons wisely.*

### **Example 1:**

**Input:** `nums = [3,1,5,8]`

**Output:** `167`

**Explanation:**

`nums = [3,1,5,8] --> [3,5,8] --> [3,8] --> [8] --> []`

`coins = 3*1*5 + 3*5*8 + 1*3*8 + 1*8*1 = 167`

### **Example 2:**

**Input:** `nums = [1,5]`

**Output:** `10`

### **Constraints:**

- `n == nums.length`
- `1 <= n <= 500`
- `0 <= nums[i] <= 100`